

PROCEEDINGS OF SPIE

[SPIDigitalLibrary.org/conference-proceedings-of-spie](https://spiedigitallibrary.org/conference-proceedings-of-spie)

The Infrared Imaging Spectrograph (IRIS) for TMT: final design development of the data reduction system

Surya, Arun, Zonca, Andrea, Rundquist, Nils-Erik, Wright, Shelley, Walth, Gregory, et al.

Arun Surya, Andrea Zonca, Nils-Erik Rundquist, Shelley A. Wright, Gregory L. Walth, David R. Anderson, Edward L. Chapin, Eric M. Chisholm, Tuan Do, Jennifer S. Dunn, Kim Gillies, Yutaka Hayano, Christopher A. Johnson, Renate Kupke, James E. Larkin, Takashi Nakamoto, Reed L. Riddle, Roger M. Smith, Ryuji Suzuki, Ji Man Sohn, Robert W. Weber, Jason L. Weiss, Kai Zhang, "The Infrared Imaging Spectrograph (IRIS) for TMT: final design development of the data reduction system," Proc. SPIE 11452, Software and Cyberinfrastructure for Astronomy VI, 114521K (17 December 2020); doi: 10.1117/12.2561766

SPIE.

Event: SPIE Astronomical Telescopes + Instrumentation, 2020, Online Only

The Infrared Imaging Spectrograph (IRIS) for TMT: Final Design Development of the Data Reduction System

Arun Surya^a, Andrea Zonca^b, Nils-Erik Rundquist^a, Shelley A. Wright^{a,c}, Gregory L. Walth^d, David Andersen^e, Edward Chapin^e, Eric Chisholm^f, Tuan Do^l, Jennifer Dunn^e, Kim Gillies^f, Yutaka Hayano^h, Chris Johnson^f, Renate Kupke^k, James Larkin^l, Takashi Nakamoto^h, Reed Riddleⁱ, Roger Smithⁱ, Ryuji Suzuki^h, Ji Man Sohn^l, Robert Weberⁱ, Jason Weiss^f, and Kai Zhang^j

^aCenter for Astrophysics and Space Sciences, University of California San Diego, La Jolla, USA

^bSan Diego Supercomputer Center, University of California San Diego, La Jolla, USA

^cDepartment of Physics, University of California San Diego, La Jolla, USA

^dCarnegie Observatories 813 Santa Barbara Street Pasadena, California, 91101 USA

^eNational Research Council of Canada - Herzberg, Victoria, BC, V9E2E7 Canada

^fThirty Meter Telescope Observatory Corporation, Pasadena, CA 91105 USA

^gPhysics & Astronomy Department, University of California Los Angeles, CA 90095 USA

^hNational Astronomical Observatory of Japan, Osawa, Mitaka, Tokyo, 181-8588 Japan

ⁱCaltech Optical Observatories, 1200 E California Blvd., Pasadena, CA 91125 USA

^jNational Astronomical Observatories / Nanjing Institute of Astronomical Optics & Technology, Chinese Academy of Sciences, Nanjing 210042, China

^kUC Observatories, 1156 High Street, Santa Cruz, CA 90605

^lPhysics & Astronomy Department, University of California Los Angeles, CA 90095 USA

ABSTRACT

IRIS (Infrared Imaging Spectrograph) is the near-infrared (0.81 μm to 2.4 μm) diffraction-limited imager and integral field spectrograph (IFS) designed for the Thirty Meter Telescope (TMT) and the Narrow-Field Infrared Adaptive Optics System (NFIRAOS). The imager will have a 34 arcsec x 34 arcsec field of view with 4 milliarcsecond (mas) sampling. The IFS consists of a lenslet array and a slicer, enabling four plate scales from 4 mas to 50 mas, with multiple gratings and filters. We will report the progress on the development of the IRIS Data Reduction System (DRS) in the final design phase. The IRIS DRS is developed in Python with the software architecture based on the James Webb Space Telescope science calibration pipeline (**stpipe**). We are developing a library of algorithms as individual Python classes that can be configured independently and bundled into pipelines. The IRIS DRS will interface with the TMT observatory software and will operate in real-time and as a stand-alone public package for offline reduction. The IRIS DRS also includes a C library for readout processing that is used for both real-time processing and post-processing. Lastly, we will discuss development of the IRIS simulation package that simulates raw spectra and imager readout-data from the Teledyne Hawaii-4RG detectors, which are used to test and develop reduction algorithms.

Keywords: ELT, instrumentation, pipeline

1. INTRODUCTION

IRIS¹ is the first-light instrument for TMT with a diffraction limited imager and an integral field spectrograph (IFS). The imager operates with a field of view (FoV) of 34 arcsecond \times 34 arcsecond using four 4k \times 4k Hawaii-4RG detectors (H4RG). The IFS works in four spatial scales, 4 milli-arcsecond (mas), 9 mas for the lenslet based

Further author information: (Send correspondence to A.S.)

A.S.: E-mail: arunsuryaoffice@gmail.com

spectrograph and 25 mas and 50 mas for the slicer based spectrograph. The instrument will operate within the wavelength range of 0.81-2.4 μm . IRIS will have 60 or more slots on the filters wheels and 14 slots on the grating turret. The gratings will operate at resolving powers of $R=4000$, 8000 and 10000. IRIS design is sequential with imager and IFS able to operate simultaneously.

The Data Reduction System (DRS)^{2,3} is a software subsystem of IRIS with routines that are used for real-time and offline processing of IRIS data to produce science-ready data products. The large number of modes of operation and the requirement of real-time reduction for the observer poses unique challenges for the IRIS DRS. IRIS will interact with other subsystems such as NFIRAOS⁴ and Telescope Control System (TCS) for its real-time operation and metadata collection. There is also need for early development of IRIS DRS to accommodate engineering activities during the long integration and testing phase. Because of the complexity of the instrument and scale of operations for TMT, IRIS DRS together with other planning tools will be essential to plan observations and produce science-ready data.

In this paper we give an overview of the current development status for IRIS DRS and new software advances made during IRIS Final Design Phase (FDP).

2. DATA REDUCTION SYSTEM ARCHITECTURE

The DRS architecture has several components to accommodate different needs of real-time and offline reduction by the astronomer.

The DRS architecture has four main components.

1. **iris_readout** is a C library which has routines for detector level processing. During an observation this library is used by the detector subsystems to have basic processing like reference pixel subtraction, non-linearity correction and sampling of the detectors readout frames. It converts the readout files in to raw science frames (see Section 3).
2. **iris_pipeline** is the Python package that implements all higher level algorithms that process the raw science frames to reduced data products for the IFS and the imager. This includes different pipelines for imager, lenslet and slicer IFS. This is also the major offline component of the DRS that observers can use to reduce data. It will include a **Cython** interface to access the **iris_readout** C Library if the user needs to execute detector level processing offline.
3. **DRS-Assembly** is the software component that is required for the real-time operation of the DRS during an observation. It receives commands and monitors events and telemetry from other TMT software subsystems like the instrument sequencer. We have currently developed a prototype for this component in Java.
4. **DRS-VIS** are the visualization tools that will provide both real-time and offline visualization of readouts, raw frames, processed 2D images and 3D IFS data. In real-time reduction it will allow astronomers to quickly evaluate the quality of observation data. We are currently planning to have these tools mainly based on publicly available tools like **CubeViz**⁵ and **DS9**.⁶

2.1 Real-time Reduction Sequence

Figure 1 shows how the DRS components are integrated into the observatory software. The **iris_readout** C library is used by detector software in the five detector controller computers, one each for the four imagers and one for the spectrograph. The detector software calls this C library for detector level processing and creates the raw frames. The output files are in FITS format and are saved in a high-capacity file system that resides on the readout computers.

During an observation the DRS-Assembly receives the information about the exposure set and the instrument configurations. The DRS-assembly software polls for the raw science frames written to the disk and will process them as they are received. The **iris_pipeline** Python processes, which are properly configured with the matching parameters for the current observing mode, will process the input raw science frames into reduced

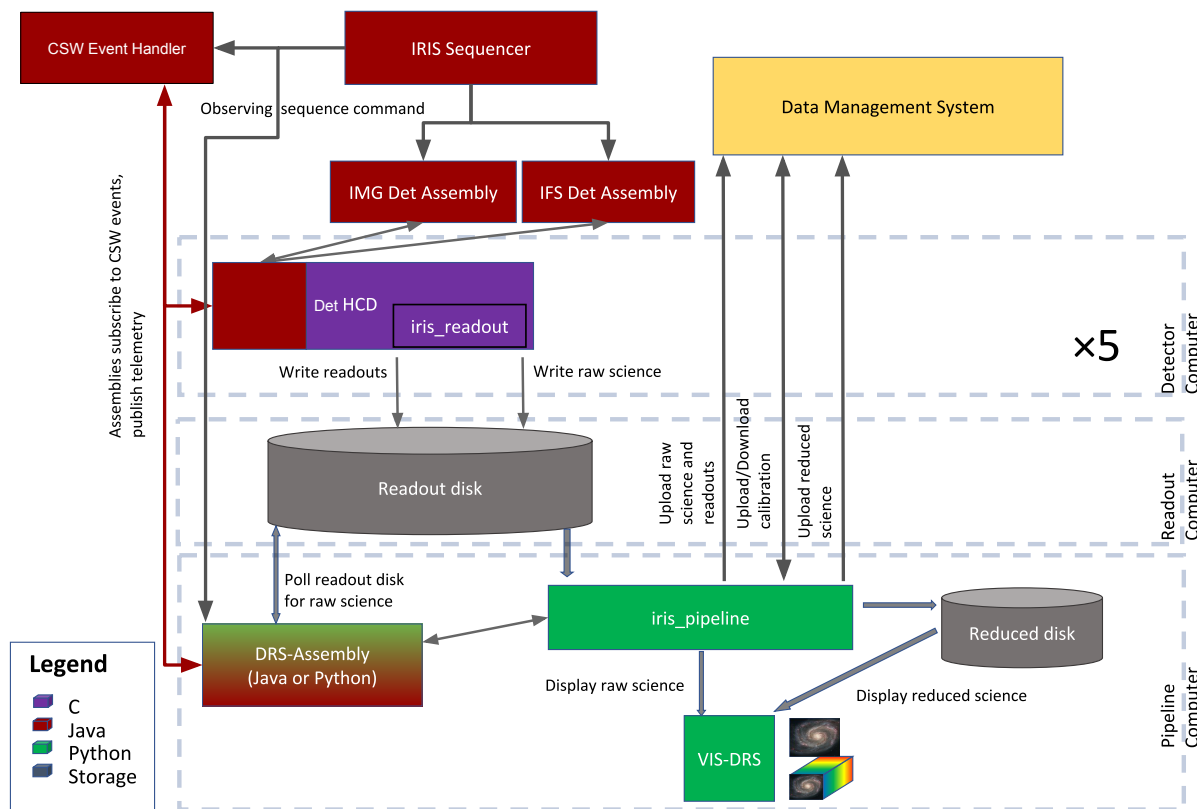


Figure 1. Architecture of the DRS-Assembly and the reduction pipelines integrated into the Observatory Software for online processing on site. The different colours are used to show the development platforms. The IRIS sequencer shares observing configuration and notifications to the DRS-Assembly. The detector software writes the readouts and processed raw frames in to the disk. DRS-Assembly invokes instances of `iris_pipeline` to process the raw frames. `iris_pipeline` also uses the interface with DMS to access calibration files and to store DRS data products. The DRS visualisation tools are used for astronomers to view the reduced and raw frames real-time during observations.

science frames in less than 15 seconds for the imagers and less than 30 seconds for the IFS. They will also connect to the TMT Data Management System (DMS) to get the necessary calibration files. The most suitable calibration files will be selected based on the metadata of the raw science frame and the pipeline configurations. The exact metadata used in each case depends on selection rules implemented in the calibration database. For example metadata information like instrument mode, filter configuration, observation time stamp, exposure time, sampling mode, subarray configuration, etc are used to in the selection rules for calibration database. There can be multiple instances of `iris_pipeline` for multiple images or spectra data. Once the pipelines have completed execution, they will write the output FITS files to the disk and also submit them for archiving into the DMS. At this point the DRS assembly can notify the end of an observation event.

2.2 Design of `iris_pipeline` package

`iris_pipeline` is the main pipeline software of DRS and will be also available for astronomers to do offline reduction of IRIS data. Together with DRS-VIS, `iris_pipeline` can be used by astronomers to process and visualize IRIS data. `iris_pipeline` can be visualized as a collection of several different pipelines working on different levels of data processing. In the DRS we define 3 levels of pipelines.

1. **Level 1** is the pipeline for detector level processing.

2. **Level 2** comprises of the instrument level pipelines specific to imager, lenslet IFS and slicer IFS. They operate on individual exposures.
3. **Level 3** comprises of the higher level pipelines that combine multiple exposures.

The basic data flow of `iris_pipeline` is shown in figure 3. For offline reduction astronomers can chose individual pipelines and customize individual steps in the pipeline. The calibration data for processing pipelines will be downloaded online through a web interface with DMS. The DMS will use the internal algorithms to fetch the right calibration files for a particular data product and pipeline configuration.

`iris_pipeline` is based upon the package `stpipe`⁷ developed by Space Telescope Institute to reduce data from both Hubble Telescope and James Webb Space Telescope. `stpipe` provides a great foundation because it has a robust and well-tested implementation of all the generic functionalities necessary to execute pipelines for image and cube data products.

`iris_pipeline` uses `stpipe` standards for:

1. Defining FITS / ASDF⁸ schemas for IRIS, which includes data, error, flags and metadata format.
2. Defining the classes which implement the IRIS algorithms. `iris_pipeline` classes inherit from the base classes in `stpipe` which gives a common template for all the algorithms and provides facilities for handling input data, calibration files, logging and other functionalities.
3. Implement configuration files for both of individual steps and of complete pipelines with text files with support for nested configurations.
4. Handle configuration for a collection of input files for pipelines through JSON files (referred to as associations).

`iris_pipeline` imports the `stpipe` package and then implements IRIS specific data models, algorithms and pipelines. Each release of `iris_pipeline` will require a specific version of `stpipe`. `iris_pipeline` is developed on Github and is already publicly available under a BSD license. It also includes a tutorial with sample data and extensive documentation *.

2.3 Data Products

DRS will save data at different levels of processing in the DMS. This data is saved in FITS format and will be available for astronomers to download and reprocess. Primarily data products processed by IRIS DRS are of three types corresponding to the levels of processing.

1. **Readout files** are the collection of individual readouts from the H4RG detectors that are read out during an exposure. These readout frames for a particular exposure are saved in a readout file. The file structure of the readout files is as below
 - **SCI [Ext 1]** : 4D datacube with individual reads. The first 2 dimensions equal the size of the detector readout. Each readout frame within a ramp is stored in the 3rd axis and multiple ramps are stored along the 4th axis.
 - **STATIC_DQ [EXT 2]** : The static 2D data quality array (bad pixel map) that applies to all readouts over all ramps.
 - **READOUT_DQ [EXT 3]**: 4D Data Quality array with flags pertaining to each individual pixel of individual readout frames within each ramp.
 - **METADATA [EXT 4]** : Binary FITS table with all the metadata corresponding to the individual reads.

The file structure is shown in figure 4.

*<https://oirlab.github.io/iris-pipeline/>

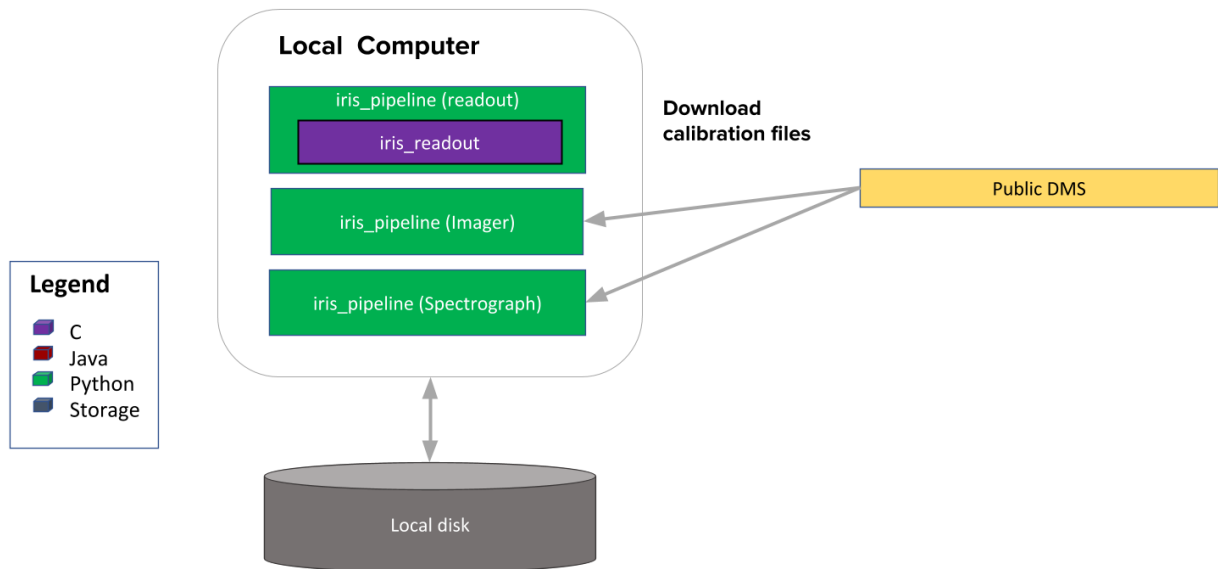


Figure 2. `iris_pipeline` offline reduction. `iris_pipeline` instances can be run with pipelines for different instrument modes and different levels of processing. Data products from a particular observation can be downloaded locally and reprocessed. The pipelines will fetch the calibration files required for a particular pipeline through an online interface with DMS. For detector level pipeline the `iris_readout` library functions will be accessible using `Cython`.

2. **Raw frames** are the frames that have been processed through detector level pipelines such as reference pixel subtraction, non-linearity correction, sampling etc. The file structure of the raw science frame is as following
 - **SCI [Ext 1]** : Raw Frame after the detector level processing.
 - **ERR [Ext 2]** : Error computed from the readout algorithms.
 - **DQ [Ext 3]** : Data Quality Map for the 2D raw frame.
 - **SUBARR_MAP [Ext 4]** : Subarray map corresponding to the exposure. The pixels corresponding to the subarrays have value of the particular subarray number.

The file structure for raw frames are shown in figure 5.

3. **Processed frames** are the files which have been processed by the different higher level pipelines and are available to astronomers as calibrated images and spectral data cubes. The file structure of the processed frames is as follows:
 - **SCI [Ext 1]** : 2D Image/3D Datacube after processing through DRS pipelines.
 - **ERR [Ext 2]** : Error computed from the pipeline algorithms.
 - **DQ [Ext 3]** : Data Quality map
 - **SUBARR_MAP [Ext 4]** : Subarray map extension that only exists for imager files.

The file structure for the processed frame for IFS data is shown in figure 6.

3. READOUT PROCESSING ROUTINES

H4RG detectors can be readout through 64 channels for the full-frame. Each of these channels has 64 pixels in a row and 4096 rows in the entire channel. In addition to the full frame mode, IRIS imager will use a subarray mode for both science purposes and as on-detector guide windows (ODGW) that will be used in the adaptive

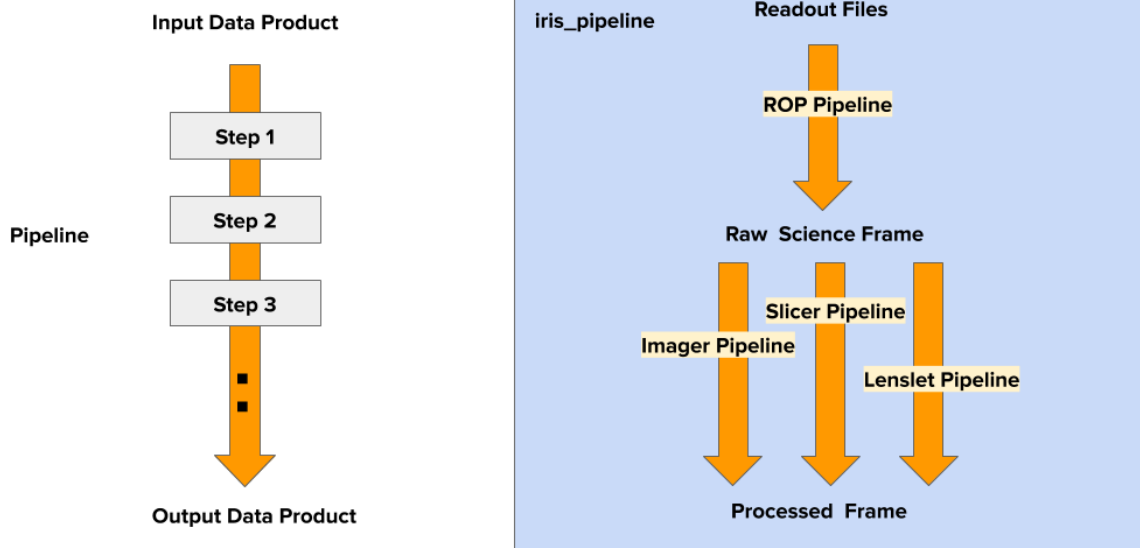


Figure 3. `iris_pipeline` data flow and file hierarchy. In `stpipe` architecture, pipeline is defined as a sequence of processing steps. The architecture allows to save intermediate data products between steps for testing and debugging. The block diagram on the right shows the data flow in `iris_pipeline`. The Readout Processing (ROP) pipeline implements the detector level processing using `iris_readout`. The output of this pipeline is passed on to instrument level pipelines depending on the mode (imager, lenslet IFS, slicer IFS) which further processes it to create final processed frame.

optics sequence. Subarrays are read using the single output mode available for H4RG that can be used to read a subset of pixels in the full-frame. The clocking pattern of the H4RG readout has to be carefully designed to incorporate subarrays. To have optimal noise performance the readout for full-frame and subarray pixels should be implemented at constant cadence. The current plan is to readout a fixed number of subarray pixels in single output mode after each row of full-frame is read out. The addition of extra subarray pixels will add overhead to the readout time of the array and need to be carefully selected. For example, a full frame of H4RG is approximately readout in 1.75s but addition of 16 subarray pixels after each row read will increase the overall read time to approximately 2.27s.

There are several routines planned to be used in the detector level processing pipelines. As discussed before most of this will be available through `iris_readout` C library. For offline processing of readouts there will be extra routines for more advanced processing. Currently following algorithms are planned for `iris_readout`.

1. **Reference Pixel Subtraction:** H4RG detectors have four rows and columns of reference pixels on the outside that are designed to mimic other light sensitive pixels. Some combination of reference pixel data is subtracted from the science pixels to remove correlated noise. For basic reference pixel subtraction algorithm, the average of reference pixels in each video output is subtracted from the science pixels in the corresponding output.
2. **Non-linearity Correction:** To correct the non-linear response of the detectors at high well-depth, non-linearity corrections are usually applied to the detector data. For the IRIS detectors non-linearity corrections will be applied using detector characterization data. The non-linearity will be modelled using a custom polynomial for each pixel in each of the five detectors.
3. **Sampling:** The H4RG detectors are read out non-destructively and each ramp of the exposure is sampled to extract the signal. For IRIS the sampling modes planned to be used are Correlated Double Sampling (CDS), Multiple Correlated Double Sampling (MCDS) and Up-the-Ramp (UTR) sampling. We have implemented these algorithms in the `iris_readout` C library.

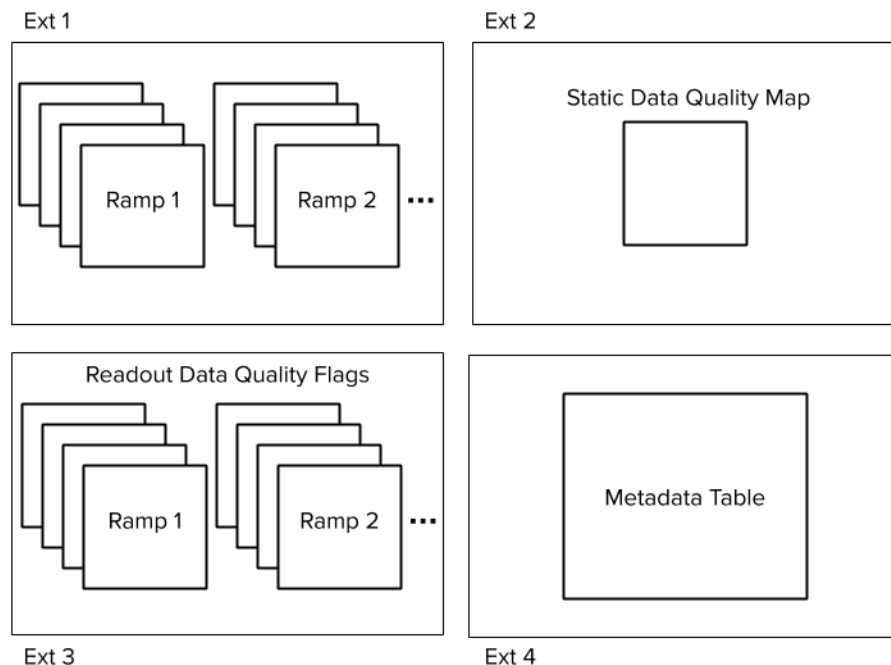


Figure 4. File structure for Readout File. The readout frames are stored as a 4D datacube in the SCI extension as a collection of ramps. Static data quality map has the 2D bad pixel map and other static flags. Readout data quality flags give the data quality flags for individual readout pixels. Finally a binary fits table is used to store the metadata for each readout frame.

4. **Jump Detection:** This routine uses the unsaturated readouts in a ramp to estimate the count rate for the ramp. Any jumps in the ramp due to cosmic ray hits or other means is detected and removed using the count rate estimates. This allows to correctly estimate the flux received in each ramp. Since this algorithm usually requires a double pass through the ramp readouts, this is implemented only in the offline readout processing.
5. **Data Quality Initialization:** This routine is used to retrieve the static data quality map from the DMS and attach it as an extension to the readout file. These data quality flags for each pixel are used by other pipeline routines during processing.

4. IMAGER AND IFS ROUTINES

Many of the imager and IFS algorithms in DRS are described in detail by Walth et al (2016)² and Walth et al.(2018).³ Currently we have completed work on prototyping some of the imager algorithms in *iris_pipeline*.

The main algorithms used in the imager pipeline are wcs assignment, dark subtraction, flat fielding, scaled sky subtraction, field distortion correction, flux calibration, mosaic/combine processed frames and persistence mitigation. The algorithms planned for the IFS pipelines are wcs assignment, dark subtraction, flat fielding, spectral extraction, wavelength calibration, cube assembly, residual ADC correction, scaled sky subtraction, flux calibration, mosaic/combine processed frames.

5. IRIS SIMULATOR SOFTWARE

IRIS simulator has been extremely useful to determine instrument sensitivities and explore a large number of science cases and technical performances of the instrument.^{9,10} A web based exposure time calculator (ETC) [†] has been also developed to assist astronomers in evaluating science cases for IRIS. The simulator uses realistic noise and background models together with simulated Point Spread Functions (PSF) from TMT and NFIRAOS.

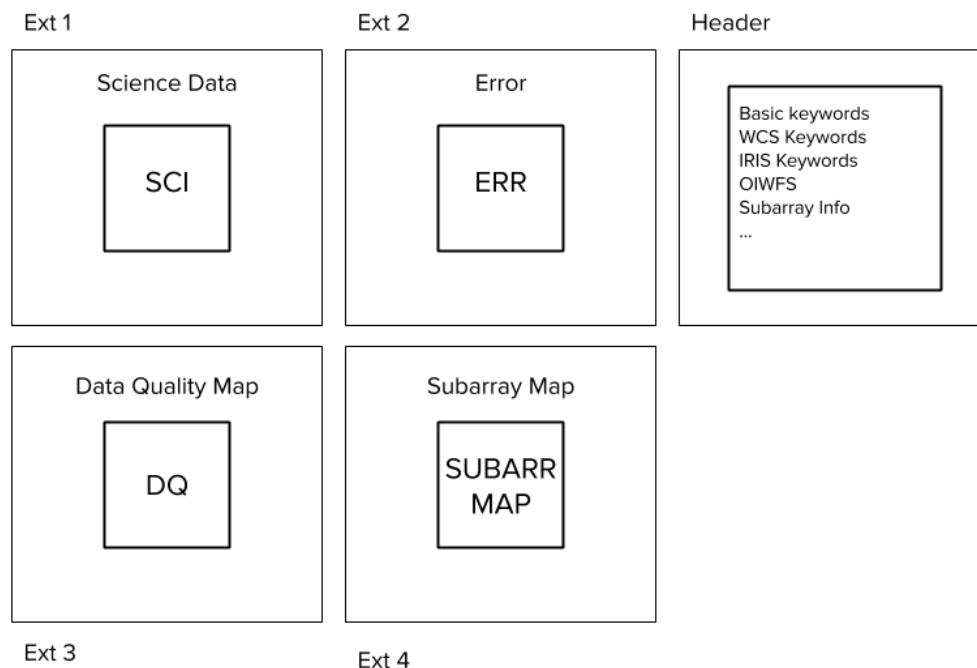


Figure 5. File structure of a Raw Science Frame. The file has four extensions: SCI for storing the 2D raw frame data, ERR to store the error from the detector level processing, DQ to store the 2D data quality flags and SUBARR_MAP to store the subarray information of the exposure in 2D.

We have developed new routines in the simulator that allows us to simulate raw data from the imager and the IFS.

These efforts has been aided by end-to-end modelling using the Zemax optical design. The spectral layout studies have also been useful in optimizing filter cutoffs and grating parameters. The following subsections discuss these new additions to the IRIS simulator package. [†]

5.1 Readout Simulation

The readout routines allow simulation of individual readouts in an exposure with realistic detector noise. These simulations are important to test the detector level processing algorithms. An important challenge for the DRS is processing of subarrays. IRIS uses subarrays to read out bright sources in the imager field to avoid saturation. There are also many science cases that would benefit from use of subarrays. Currently the readout simulator routines are capable of simulating observations involving subarrays.

For simulating realistic detector noise from H4RG detectors we have used the `ngxrg`¹¹ library developed for JWST. `ngxrg` library simulates realistic correlated and uncorrelated noise in Teledyne HXRG detectors. The readout data products from these simulations are used for testing and prototyping the `iris_readout` C library.

5.2 IFS raw spectra simulation

The iris simulator simulates raw spectra from slicer and lenslet IFS. To simulate raw spectra the end-to-end optical model of the spectrographs in Zemax are used. The spectral trace from the Zemax models are combined with PSF information to produce the realistic raw spectra. These simulations also include detector level noise and sky background.

The large number of IFS modes with varying FoV, gratings and filters are included in the raw spectra simulation. An example of the raw slicer spectra simulation is shown in figure 8. The raw spectra simulations

[†]<https://www.tmt.org/etc/iris>

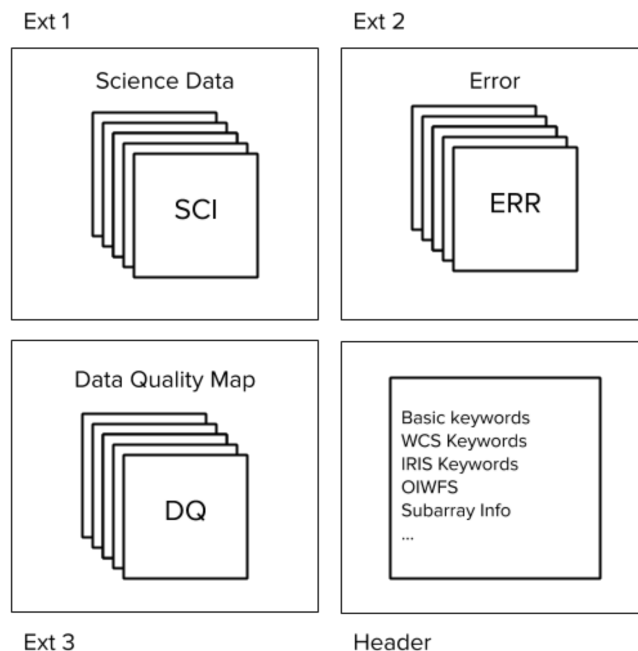


Figure 6. File structure for a processed IFS file. Its structure is similar to raw science frame. It has three main extensions SCI, ERR and DQ which are used to store science data, error values and data quality flags respectively. For the IFS processed frame these data are in the shape of 3D data cubes.

for lenslet spectrograph (figure 9) use the Zemax optical model of the instrument to generate the spectral layout in the detector. These spectral traces are used together with the physical optics simulation of the micro pupils to simulate the spectra at the detector. The detector image plane is sampled at subpixels to accurately model the spectral cross-talk for the closely spaced 16384 spectra from the 128 x 128 spaxels. Finally the flux generation and noise modelling routines are used to generate a realistic spectra.

The lenslet IFS with finest plate scale of 4 mas/spaxel aims to preserve the superb image quality supplied by the adaptive optics system. The lenslet array is designed with a pitch of 350 microns and creates an array of micro pupil images that will be dispersed by the spectrograph. These micro pupil images are dominated by diffraction effects and create spots larger than the geometric pupil size estimated by ray tracing. This impact of sub-sampling of the adaptive optics corrected point spread functions has to be modeled using physical optics propagation. We use Zemax physical optics simulations to accurately model such effects. Such simulations are very important to model cross-talk between the closely spaced spectra of the lenslet spectrograph.

5.3 Calibration Data

The raw data simulations also use spectra from lamp spectra for wavelength calibration and flat fielding. For wavelength calibration Xe, Ar and Kr lamps are planned to be used. Continuum sources are used for flat fielding. We have used publicly available line lists for these lamps and modelled it for raw data simulator according to the resolving power and sampling. The simulated calibration frames are shown in figure 10 and 11.

6. SUMMARY

IRIS DRS design faces unique challenges owing to the large number of instrument modes available in IRIS and the need for cohesive functioning together with other subsystems like NFIRAOS. Lot of advances has been made to the IRIS DRS design during the final Design Phase. A major update to the design and prototyping effort has been the adoption of **stpipe** architecture for the DRS. We have also made progress developing routines for simulating various data products used by IRIS DRS at different stages for processing. This has been possible due

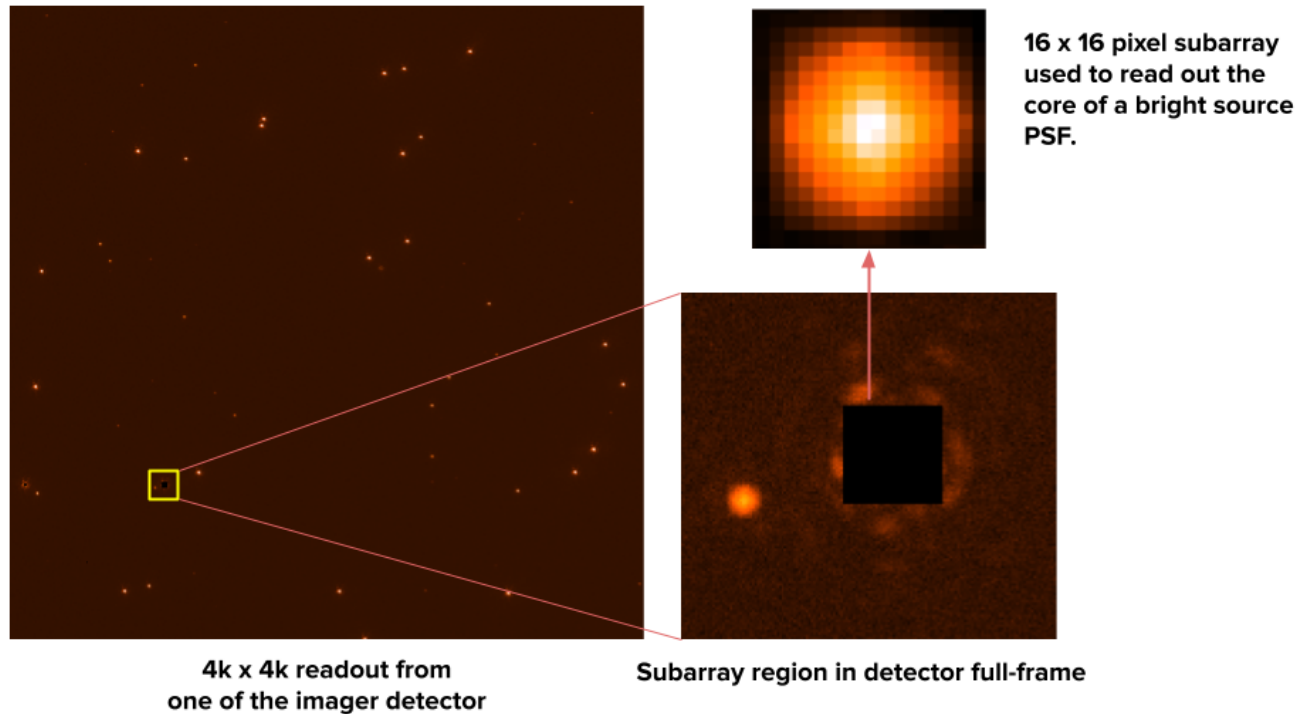


Figure 7. Simulated subarray and full frame with readout simulator. Subarrays are sub-regions that can be readout faster than the entire array. This gives the ability to read out bright sources in the field that will saturate faster. The figure shows a simulated 16×16 subarray readout of the core of a bright source that saturates. This subarray would be read out every 4 milliseconds compared to the full-frame of the detector which will be read out approximately every 2.27 considering the 16 subarray pixel per row scheme. The DRS processes subarray data separately through the detector level pipelines. The higher level pipelines will combine the subarray data in to the image frame during processing and visualisation.

to the new routines in IRIS simulator that uses end to end modelling using Zemax. DRS will be an important subsystem for IRIS which will be critical from early integration and testing stages and will continue to deliver science ready data after first light.

ACKNOWLEDGMENTS

The TMT Project gratefully acknowledges the support of the TMT collaborating institutions. They are the California Institute of Technology, the University of California, the National Astronomical Observatory of Japan, the National Astronomical Observatories of China and their consortium partners, the Department of Science and Technology of India and their supported institutes, and the National Research Council of Canada. This work was supported as well by the Gordon and Betty Moore Foundation, the Canada Foundation for Innovation, the Ontario Ministry of Research and Innovation, the Natural Sciences and Engineering Research Council of Canada, the British Columbia Knowledge Development Fund, the Association of Canadian Universities for Research in Astronomy (ACURA), the Association of Universities for Research in Astronomy (AURA), the U.S. National Science Foundation, the National Institutes of Natural Sciences of Japan, and the Department of Atomic Energy of India.

REFERENCES

- [1] Larkin, J. E., Moore, A. M., Wright, S. A., Wincentsen, J. E., Anderson, D., Chisholm, E. M., Dekany, R. G., Dunn, J. S., Ellerbroek, B. L., Hayano, Y., Phillips, A. C., Simard, L., Smith, R., Suzuki, R., Weber,

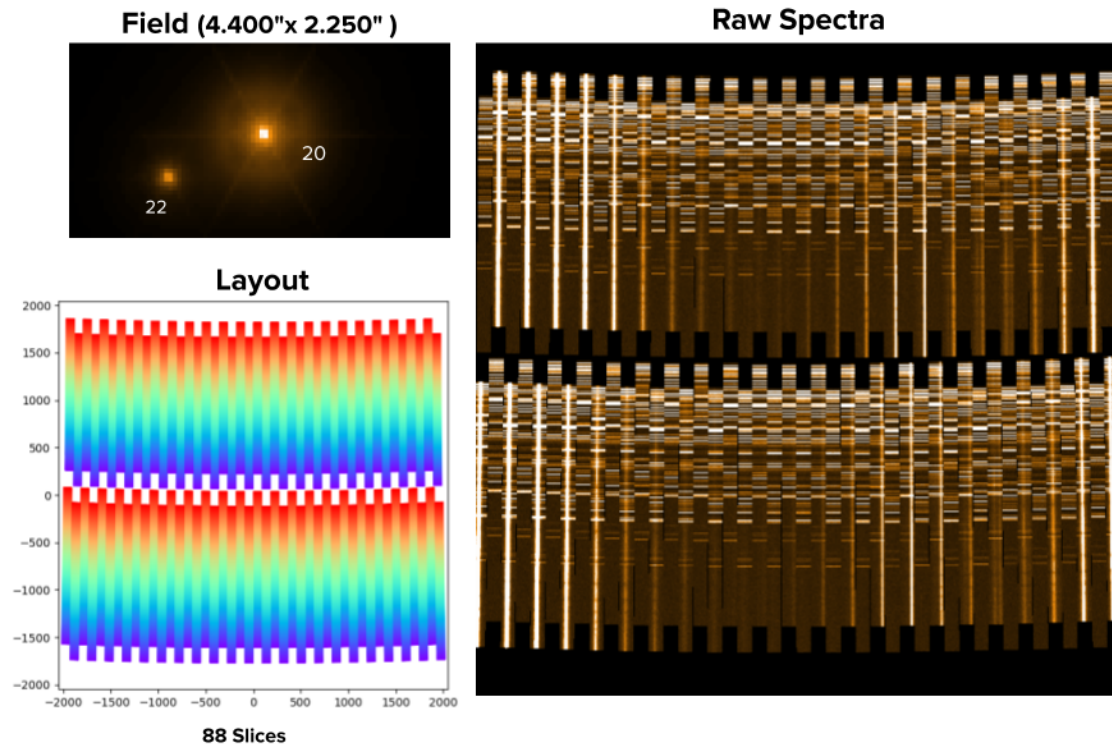


Figure 8. Raw spectra simulation of the slicer IFS. The figure shows the simulation of a binary star observed using the slicer IFS. The simulation was done for Zbb filter ($0.81\text{--}0.99\ \mu\text{m}$) with $R=4000$ at 50 mas scale of the slicer. The field and the spectral layout of the slicer used for the simulations is shown on the left. The raw spectra simulated is shown on the right.

- R. W., Weiss, J. L., and Zhang, K., “The Infrared Imaging Spectrograph (IRIS) for TMT: instrument overview,” in [*Ground-based and Airborne Instrumentation for Astronomy VI*], Evans, C. J., Simard, L., and Takami, H., eds., *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series* **9908**, 99081W (Aug. 2016).
- [2] Walth, G., Wright, S. A., Weiss, J., Larkin, J. E., Moore, A. M., Chapin, E. L., Do, T., Dunn, J., Ellerbroek, B., Gillies, K., Hayano, Y., Johnson, C., Marshall, D., Riddle, R. L., Simard, L., Sohn, J. M., Suzuki, R., and Wincentzen, J., “The Infrared Imaging Spectrograph (IRIS) for TMT: data reduction system,” in [*Software and Cyberinfrastructure for Astronomy IV*], Chiozzi, G. and Guzman, J. C., eds., *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series* **9913**, 99134A (Aug. 2016).
- [3] Walth, G. L., Wright, S. A., Rundquist, N.-E., Andersen, D., Chapin, E., Chisholm, E., Do, T., Dunn, J., Ellerbroek, B., Gillies, K., Hayano, Y., Johnson, C., Larkin, J., Nakamoto, T., Riddle, R., Simard, L., Smith, R., Suzuki, R., Sohn, J. M., Weber, R., Weiss, J., and Zhang, K., “The Infrared Imaging Spectrograph (IRIS) for TMT: advancing the data reduction system,” in [*Software and Cyberinfrastructure for Astronomy V*], Guzman, J. C. and Ibsen, J., eds., *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series* **10707**, 1070731 (July 2018).
- [4] Crane, J., Herriot, G., Andersen, D., Atwood, J., Byrnes, P., Densmore, A., Dunn, J., Fitzsimmons, J., Hardy, T., Hoff, B., Jackson, K., Kerley, D., Lardière, O., Smith, M., Stocks, J., Véran, J.-P., Boyer, C., Wang, L., Tranco, G., and Trubey, M., “NFIRAOS adaptive optics for the Thirty Meter Telescope,” in [*Adaptive Optics Systems VI*], Close, L. M., Schreiber, L., and Schmidt, D., eds., *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series* **10703**, 107033V (July 2018).
- [5] Martin, M., Abicht, K., Stadler, C., Ngonga Ngomo, A.-C., Soru, T., and Auer, S., “Cubeviz: Exploration and visualization of statistical linked data,” in [*Proceedings of the 24th International Conference on World*

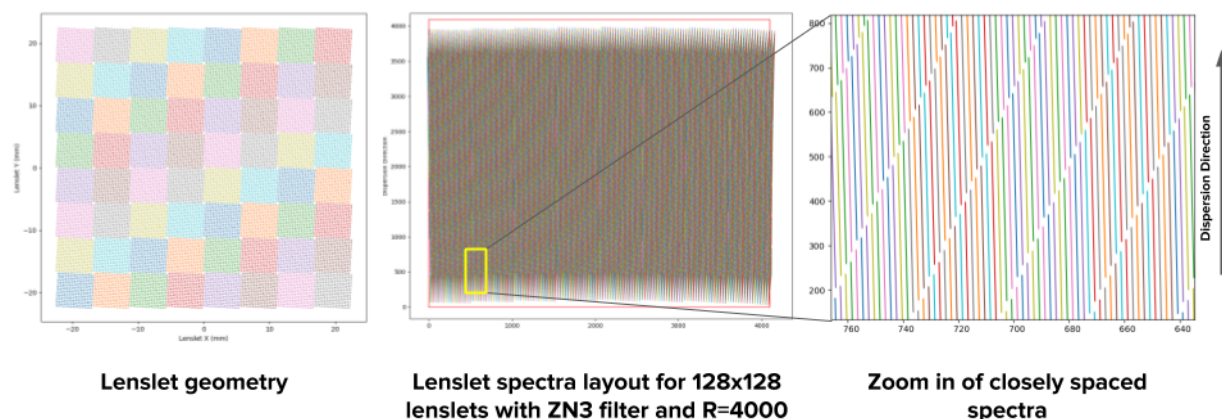


Figure 9. Spectral layout for the lenslet IFS. The simulation used ZN3 filter ($0.89\text{--}0.94\ \mu\text{m}$) with $R=4000$ with 4mas spatial scale. [Left] Geometry of the 128×128 lenslet array used in IRIS. The individual 16×16 groups are staggered from each other and the lenslet array is tilted by a small angle to avoid the overlapping of spectra from adjoining lenslets in the same row. In the figure 16×16 lenslet groups are coloured separately to visualize the staggered pattern of the array. [Center] The spectral layout of lenslet. [Right] Zoomed in view of the closely separated spectra from the lenslet IFS. Spectra from different lenslets are coloured differently. The spectra are separated by $30\ \mu\text{m}$ on the detector plane.

Wide Web], *WWW '15 Companion*, 219222, Association for Computing Machinery, New York, NY, USA (2015).

- [6] Joye, W. A. and Mandel, E., “New Features of SAOImage DS9,” in [*Astronomical Data Analysis Software and Systems XII*], Payne, H. E., Jedrzejewski, R. I., and Hook, R. N., eds., *Astronomical Society of the Pacific Conference Series* **295**, 489 (Jan. 2003).
- [7] Bushouse, H., Eisenhamer, J., and Davies, J., “The JWST Data Calibration Pipeline,” in [*Astronomical Data Analysis Software and Systems XXVII*], Teuben, P. J., Pound, M. W., Thomas, B. A., and Warner, E. M., eds., *Astronomical Society of the Pacific Conference Series* **523**, 543 (Oct. 2019).
- [8] Greenfield, P., Droettboom, M., and Bray, E., “Asdf: A new data format for astronomy,” *Astronomy and Computing* **12**, 240 – 251 (2015).
- [9] Wright, S. A., Barton, E. J., Larkin, J. E., Moore, A. M., Crampton, D., and Simard, L., “The infrared imaging spectrograph (IRIS) for TMT: sensitivities and simulations,” in [*Ground-based and Airborne Instrumentation for Astronomy III*], McLean, I. S., Ramsay, S. K., and Takami, H., eds., *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series* **7735**, 77357P (July 2010).
- [10] Wright, S. A., Walth, G., Do, T., Marshall, D., Larkin, J. E., Moore, A. M., Adamkovics, M., Andersen, D., Armus, L., Barth, A., Cote, P., Cooke, J., Chisholm, E. M., Davidge, T., Dunn, J. S., Dumas, C., Ellerbroek, B. L., Ghez, A. M., Hao, L., Hayano, Y., Liu, M., Lopez-Rodriguez, E., Lu, J. R., Mao, S., Marois, C., Pandey, S. B., Phillips, A. C., Schoeck, M., Subramaniam, A., Subramanian, S., Suzuki, R., Tan, J. C., Terai, T., Treu, T., Simard, L., Weiss, J. L., Wincentzen, J., Wong, M., and Zhang, K., “The infrared imaging spectrograph (IRIS) for TMT: latest science cases and simulations,” in [*Adaptive Optics Systems V*], Marchetti, E., Close, L. M., and Véran, J.-P., eds., *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series* **9909**, 990905 (July 2016).
- [11] Rauscher, B. J., “Teledyne H1RG, H2RG, and H4RG Noise Generator,” *pasp* **127**, 1144 (Nov. 2015).

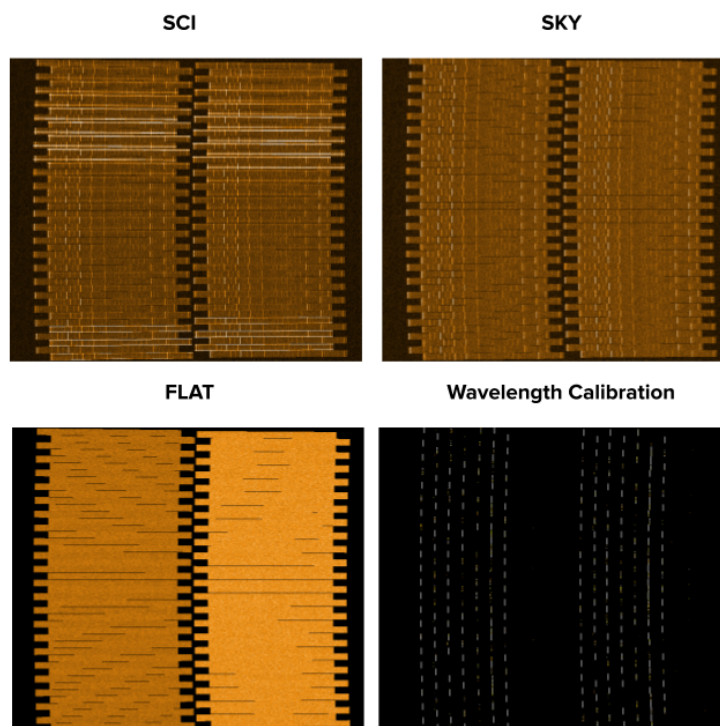


Figure 10. Simulated raw frames of slicer IFS including science and calibration frames. Currently the simulator can generate science frames and calibration frames including sky, flat and lamp spectra for wavelength calibration.

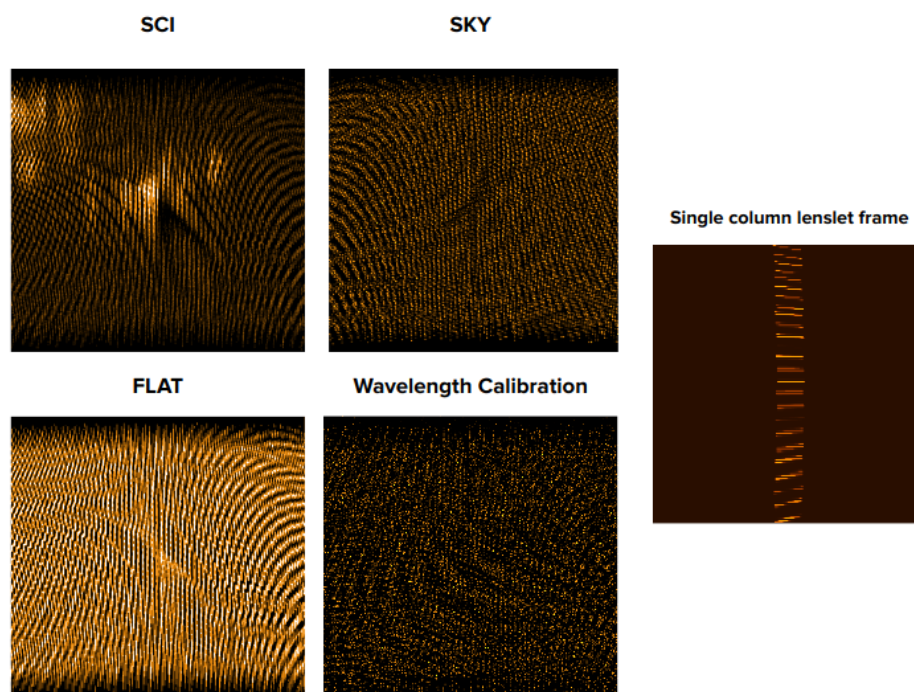


Figure 11. Simulated raw frames of lenslet IFS. This includes science frames, calibration frames and engineering frames. Currently the simulator generates sky, flat and lamp spectra for wavelength calibration. The simulator also generates single column flat spectra to generate rectification matrices to assist in deconvolution routines of the lenslet IFS pipeline.